

Requirements Engineering as Creative Problem Solving: A Research Agenda for Idea Finding

Neil Maiden, Sara Jones, Kristine Karlsen, Roger Neill, Konstantinos Zachos, Alastair Milne
Centre for Creativity in Professional Practice,

City University London
London EC1V 0HB, UK

n.a.m.maiden@city.ac.uk, saraj@soi.city.ac.uk, kristine.karlsen@city.ac.uk, rogern@perdiemprojects.com,
kzachos@soi.city.ac.uk, alastairmilne@gmail.com

Abstract—This vision paper frames requirements engineering as a creative problem solving process. Its purpose is to enable requirements researchers and practitioners to recruit relevant theories, models, techniques and tools from creative problem solving to understand and support requirements processes more effectively. It uses 4 drivers to motivate the case for requirements engineering as a creative problem solving process. It then maps established requirements activities onto one of the longest-established creative problem solving processes, and uses these mappings to locate opportunities for the application of creative problem solving in requirements engineering. The second half of the paper describes selected creativity theories, techniques, software tools and training that can be adopted to improve requirements engineering research and practice. The focus is on support for problem and idea finding – two creative problem solving processes that our investigation revealed are poorly supported in requirements engineering. The paper ends with a research agenda to incorporate creative processes, techniques, training and tools in requirements projects.

Keywords—Requirements engineering; creativity

I. REQUIREMENTS ENGINEERING AS CREATIVE PROBLEM SOLVING

We have reported creativity research and practice in requirements engineering for almost a decade now, for example in [32, 33]. Most of our papers have described a new case study that demonstrates the use of creativity techniques [34] and/or tools in a requirements project [30]. Whilst we believe that the concrete results reported in these papers have been useful to researchers and practitioners, it has meant that we have not presented our wider vision for creativity in requirements engineering – not until now, that is. In this vision paper we motivate and outline an agenda for requirements engineering research and practice framed as creative problem solving, and propose that requirements engineering communities recruit and exploit the established body of knowledge on creative problem solving.

Our vision is underpinned by 2 important assertions that we seek to provide evidence for in the paper. The first is that many requirements engineering processes are creative problem solving processes. We believe, however, that the emphasis on the *engineering* of requirements and software has marginalized creative thinking as an important

requirements activity. Therefore we mapped requirements processes onto one established creative problem solving method to demonstrate how creative problem solving processes incorporate current requirements processes. The second assertion is that many requirements processes can be enhanced to produce more useful and novel outcomes if treated as creative problem solving. These outcomes can include new system and product concepts, requirements made possible by new technologies, changes to system boundaries to widen requirements opportunities, and new combinations of requirements. To support our assertion we report creativity techniques, software tools and training that are unfamiliar to most requirements engineers but could be applied in their projects.

Creativity has been the subject of research in different disciplines for much longer than research in software and requirements engineering. The Greek philosophers considered the nature of creativity in human endeavor. In the late 19th century mathematician Henri Poincare reflected on the nature of successful creative thinking in science [23]. Creativity research in its current form was born during the 2nd world war in the U.S. military, and grew in the 1950s when Osborn and Parnes [48] and Synectics [20] developed new creative solving processes. During the 1960s and 1970s leaders such as Edward De Bono developed lateral thinking [11] and Genrich Altshuller evolved the TRIZ method (in [55]) for structured creative problem solving. More recent creativity research has been undertaken in disciplines including cognitive psychology [9], artificial intelligence [51] and product design [40]. The result, today, is a large and multidisciplinary body of knowledge of theories and models, and large collections of processes, techniques and tools for creative problem solving.

We believe that requirements engineering researchers and practitioners can benefit from this body of knowledge about creative problem solving. One role of this vision paper is to encourage the exploration and transfer of theories, knowledge and techniques from creative problem solving research to requirements engineering. By framing requirements engineering as creative problem solving we can gain new insights into it, and recruit new knowledge from other disciplines to understand it better and support it more effectively.

Throughout this paper we adopt Sternberg's definition [57] as prototypical of those reported. Creativity is defined as "*the ability to produce work that is both novel (i.e. original, unexpected) and appropriate (i.e. useful, adaptive to task constraints)*". Using this definition we expect creative requirements processes to produce requirements and other artifacts that are novel, useful and adapted to the constraints on requirements processes.

But what is novel? Boden [3] distinguishes *historically* creativity, which creates something novel to person-kind, from *psychological* creativity that is new to the person but not to others. Suwa [58] refined this distinction by introducing *situated* creativity, in which a designer or reasoner has an idea for a specific task novel in that particular situation or domain. Most creativity in requirements projects is a form of *situated* creativity, creating requirements and other outcomes *new to project stakeholders but not historically new*. We also distinguish between creativity and innovation. Creativity focuses on the generation of novel and useful ideas, whilst we define innovation as the practical application of the results from creativity thinking – the *creativity that ships* described by Bill Gates. In this paper we associate innovation more with downstream software development processes, and focus on *situated* creativity in requirements processes.

The remainder of the paper is in seven sections. The next reports four drivers for framing requirements engineering as creative problem solving. Section III reports mappings between creative problem solving and requirements engineering processes that provide evidence for requirements engineering as creative problem solving. Sections IV to VII report creativity theories and models with which to frame creativity techniques, tools and training that could be exploited in requirements engineering. The paper ends with a research agenda that extends requirements processes with creative idea finding mechanisms.

II. DRIVERS FOR MORE CREATIVITY IN REQUIREMENTS PROJECTS

The last few years have seen the emergence of different external drivers for framing requirements engineering as creative problem solving. Some are macro-economic, others trends in software development methods. In this paper we present four of these drivers.

A. The Strategic Importance of Creativity and Innovation

The current strategic importance of creativity and innovation has been acknowledged by many commentators, both at the international level – the Nomura Institute argued that "*Creativity will be the next economic activity, replacing the current focus on information*" – and at national level such as within the UK, where the Cox review commissioned by Gordon Brown in 2005 saw exploitation of the nation's creative skills as *vital to the UK's long-term economic success* [8]. The review concluded that *the success of the*

creative industries notwithstanding, there is evidence that UK business is not realizing the full potential of applying creativity more widely. Brown added that the challenge is *not just to encourage creative industries, our priority is to encourage all industries to be creative*. Creativity and design, used effectively, are important competitive tools for firms and the formal education system, enhancing the supply of creativity and design skills, and management and business skills more generally [13] in the UK and other industrialized countries.

In this paper we argue that requirements processes need to align more effectively with contemporary business processes in being more innovative and supportive of creative thinking if these processes are to deliver competitive advantage. Such competitive advantage is even more important in difficult economic times.

B. A Wider Range of Solution Technologies

Until a decade ago the number of technologies available to implement computer-based systems was limited. However, since then, we have seen a substantial increase in the range of technologies, devices and applications available, and this increase will continue in the foreseeable future. Moreover, information about these devices and applications is more easily accessible to requirements processes via the Internet, using Google searches and app stores from organizations such as Apple [1] and Nokia [43]. The importance of these existing technologies and applications as a source of requirements should not be under-estimated. Nye et al. [45] reported that necessity is often *not* the mother of invention, and that software tools often exist before the problems to be solved, an argument supported by Norman [44] who describes how technologies have often preceded user needs which only emerge later.

If future software development is to embrace and support new opportunities made possible by the plethora of new devices and applications, then requirements processes will need to incorporate these devices and applications in order to create requirements from them.

C. The Growth in Agile Development Methods

Software development has seen a substantial growth in the use of agile development methods and techniques over the last decade [6]. Although most agile methods and techniques do not explicitly support creativity and creative thinking, the focus on developing working code in short cycles means that project teams explore requirements and software solutions in parallel. Effective communication between end-user stakeholders and software developers can lead not only to new solution innovations in response to user requirements, but also novel requirements in response to exploratory interactions with the software. Current agile methods tend not to exploit working software for creative thinking about new requirements and opportunities. Indeed the short durations of sprints can discourage the incubation and reflection needed for creative thinking. However, as

agile methods evolve and new technologies with which to develop solution emerge, we foresee more opportunities to discover novel requirements from working software.

D. Trends in Requirements Research and Practice

Relatively little requirements engineering research has addressed creativity. RAD workshops [15] make tangential reference to creative problem solving. Robertson [52] argues that requirements analysts need to be inventors to bring about the innovative change in a product or business that gives competitive advantage, and Nguyen et al. [46] observed that teams restructured requirements models at critical points when triggered by moments of sudden insight. However, few researchers have developed software tools to support creative thinking during requirements processes, and requirements engineering has yet to deliver accepted tools that can support creative thinking.

In contrast, requirements practitioners have become increasingly interested in creativity. Participation in creativity tutorials has increased substantially recently, and practitioners are seeking new means to create competitive advantage through software solutions for the reasons reported for the first two drivers. We believe that requirements research needs to reduce the gap between research state-of-the-art and these practitioner needs.

III. WHY REQUIREMENTS ENGINEERING CAN BE FRAMED AS CREATIVE PROBLEM SOLVING

There is no need to define requirements engineering for the purposes of this vision paper – numerous adequate definitions are available [27, 65]. On the other hand it is useful to return to the requirements engineering roadmap reported in [47] to explore different theories and disciplines that underpin requirements research and practice. Their analysis is admirably broad. As well as recognizing the importance of computer science, logic and systems engineering, the roadmap lists cognitive psychology, anthropology, sociology, linguistics and philosophy as having both theoretical and practical relevance for requirements work.

However the analysis also reveals some possible limitations of their scoping of requirements research and practice. For example, their reporting of cognitive psychology addresses problems of tacit knowledge and matching to user models – problems that suggest a process of eliciting and modeling rather than creating requirements. One reason is that Nuseibeh & Easterbrook frame the requirements problem using already-established processes – eliciting, modeling, analyzing, communicating, agreeing and evolving requirements. They appear to have viewed requirements engineering as processes of problem scoping and understanding rather than problem solving.

Therefore, to demonstrate the importance of creativity in requirements engineering, we framed established requirements processes using one of the original creativity problem solving processes – the *CPS method* developed by

Osborn and Parnes [48]. Many subsequent creative problem solving methods such as *Synectics* [19] are based on its structure. The method was originally intended to help people understand and use their creative talent more effectively. It supports six stages of problem solving: *objective finding*, *fact finding*, *problem finding*, *idea finding*, *solution finding* and *acceptance finding*. The six stages are arranged into three groups – *understanding the problem* (stages 1 & 2), *idea generation* (3 & 4), and *planning for action* (5 & 6).

We mapped each of the 6 stages of the CPS method onto existing software development processes. Results are shown in Table 1. Stages 5 and 6, planning for action, map more effectively onto the downstream design and implementation phases of software development, and are not considered further in this vision paper. In contrast the first four stages for understanding the problem and generating ideas were mapped onto indicative requirements processes and techniques listed in Table 1: the list is not complete.

Table 1: Mappings between stages of the CPS Method and established software development processes

Osborn-Parnes process stages	Software development processes
1. Objective finding: brainstorm a list of objectives and goals	Goal modeling [van Lamsweerde 2008], Requirements mining [Goldin & Berry 1997]
2. Fact finding: brainstorm all facts related to each objective for all perspectives and participants	Requirements acquisition [Maiden & Rugg 1996], modeling languages [Greenspan et al. 1994], satisfaction arguments [Hammond et al. 2001]
3. Problem finding: brainstorm different ways to express the problem	Goal modelling [Yu & Mylopoulos 1994], requirements expression [Robertson & Robertson 1999]
4. Idea finding: divergent brainstorming writing down all ideas, then convergence and selecting relevant ideas	Requirements decomposition and refinement [van Lamsweerde 2008], architecture patterns [Buschmann et al. 1996]
5. Solution finding: Take action on selected ideas, and develop solution selection criteria	Architecture modeling, and use of non-functional requirements for selection criteria [Franch & Maiden 2003]
6. Acceptance finding: consider real-world issues to overcome, implementation issues	Systems design and implementation, down-stream project planning

The first stage, objective finding, seeks to establish objectives and goals relevant to a new problem. Establishing the objectives and goals of a new system is also important requirements processes currently supported by techniques such as goal modeling [61], scenario analyses [59] and mining requirements from documents [18]. We argue that objective finding in creative problem solving and requirements processes are similar: both are invoked at the start of projects to discover what the project outcomes are expected to achieve. Indeed, perhaps because of the nature of software-based systems, requirements engineering has developed more sophisticated techniques for finding

objectives and goals for new systems. Therefore, we believe that there is little scope to exploit objective finding techniques from creativity research in requirements engineering.

The second stage, fact finding, seeks to discover other types of knowledge needed during problem solving. Fact finding has also been the subject of extensive requirements research. The ACRE framework [35] proposes different knowledge acquisition techniques to use, whilst techniques such as ethnographic studies [28] and cultural probes [17] recruited from other disciplines stress the importance of fact finding in early requirements processes. Requirement researchers advocate knowledge representation languages such as TROPOS to express and reason about domain facts [7, 22], and Michael Jackson's [26] distinction between the machine and domain led to techniques such as satisfaction arguments that necessitate the surfacing and documentation of domain assumptions in requirements specifications [24]. We argue that fact finding processes in requirements and creative problem solving processes are also similar, and again requirements research has already developed and recruited techniques to find facts from other disciplines. We believe that there is little scope to exploit creativity fact finding techniques in requirements engineering.

The third stage, problem finding, encourages exploration of a problem by quickly expressing it in different ways in order to provide different insights into it. Requirements engineering offers numerous notations, from different types of goal models to scenarios and rich pictures to express problems from different perspectives. However, the skill and effort needed to use these notations usually mean that analysts tend to use only one or two in a typical project. Indeed many of these notations are what Green calls viscous [20] and not quick to develop or change. We believe that this viscosity can inhibit effective problem finding through pain-free exploration. Recent studies of the cognitive effectiveness of the i^* goal modeling notation still focus on model comprehension [41] rather than problem finding as a primary role of a requirements model. Few requirements notations have been developed explicitly to explore effective problem finding. Therefore we believe that there is scope to exploit problem finding techniques from creativity research in requirements engineering.

The fourth stage, idea finding, seeks to generate as many ideas as possible. In requirements projects ideas are expressed as concepts, functional requirements and design features, and requirements research has developed different techniques and software tools with which to discover new requirements. Many use pre-defined rules to search a space of possible requirements. For example the ART-SCENE tool automatically generates alternative course events in scenarios using pre-defined classes of abnormal behaviour [37], the KAOS method applies pre-defined obstacles to the attainment of goals [61], and EasyWinWin uses pre-defined techniques to find win-conditions for stakeholders [4]. Even requirements acquisition techniques such as structured

interviews and laddering [35] provide pre-defined probes that lead the analyst to ask questions that are designed primarily to search the space of known requirements. The expected outcome from each of these techniques, we argue, is to deliver a more complete set of *useful* requirements rather than deliver more *novel* requirements – the pre-defined rules describe all possible alternative course events in a scenario and all possible obstacles to the attainment of a goal, not creative prompts with which to find novel requirements. Therefore, we believe that there is scope to extend existing requirements searching techniques with idea finding from creativity research to enable the generation of useful and novel requirements. Indeed, these search techniques can be used to discover novel design ideas and detailed features in the space of possible software solutions.

Of course, other requirements processes such as requirements modeling, verification and management cannot be framed as creative problem solving. Therefore these are beyond the scope of the analysis in this paper.

This exercise revealed that requirements engineering can be framed as creative problem solving. Our mapping of established requirements processes onto the stages of the CPS method revealed that requirements support for problem and idea finding is relatively weak, and there is potential to import established creativity research and practice into the requirements discipline to support problem and idea finding. Therefore, in the remainder of the paper, we report relevant creativity theories, techniques, software tools and training for generating ideas and their application in requirements engineering.

IV. APPLYING CREATIVITY MODELS TO REQUIREMENTS ENGINEERING

Creativity research has produced many definitions, theories and models of creativity and creative problem solving process. In this section we report some of this research that, we argue, can be applied to provide new descriptive theories and models of requirements engineering. In particular we report a taxonomy of creativity categories, a definition of novelty, incubation versus illumination, and different types of creative process. Moreover we link these taxonomies, definition and processes to requirements theories, models and practices.

A. Different Types of Creativity

Margaret Boden distinguished between two basic categories of creativity: exploratory creativity and transformational creativity [3]. One explored a search space of partial and complete possibilities – what Boden called exploratory creativity. The existence of a space, she argues, implies the existence of certain rules which define the space. If they exist, then, presumably, those rules could be changed, producing what might be thought of as a paradigm shift. This kind of change was called transformational creativity. Hence, Boden argued that an idea or artifact was

novel if it did not occur before, but was reached by an existing set of generative rules.

We conjecture that requirements engineering can also be viewed as a process of exploratory creativity – a process that searches spaces of partial and complete possibilities. Methods and techniques such as KAOS, ART-SCENE and laddering reported in Section III all use pre-defined generative rules to search spaces of possible requirements, although these rules were designed to find complete rather than novel requirements. An example of pre-defined rule in ART-SCENE is [*Specify a requirement to handle the failure of a system actor in a scenario event*].

If we extend the set of pre-defined rules to generate requirements that stakeholders would define as novel as well as complete, then we can describe the resulting requirements process as one of exploratory creativity – reaching a novel idea with an existing set of generate rules.

What is less common in requirements engineering is to change search space rules to lead to transformational creativity. Transformational creativity is most likely the category of creativity described by Nye et al. [45] in their descriptions of tools and technologies that exist before the problems that these tools eventually solve – tools and technologies such as the *iPhone* and *Google Maps*. To deliver such advances in individual projects the requirements process must change the rules defining the space of possible requirements. If we assume that requirements are phenomena shared between the domain and the machine [26], then the space changes if predefined rules about the domain, the machine and/or interaction between the domain and the machine change. To change rules about the domain a requirements process must challenge domain assumptions, boundaries and laws. To change rules about the machine the process must challenge assumptions about the nature and boundaries of software-based solutions. And to change rules about domain-machine interaction, the process can challenge existing interactions. If, as a result, a requirements process can change the rules to generate requirements that stakeholders would define as novel as well as complete, then we can describe the process as one of transformational creativity – reaching a novel idea by deliberately changing an existing set of generative rules.

Furthermore Boden identifies one specific form of exploratory creativity called combinational creativity, which is the process of making unfamiliar connections between familiar items in the pre-defined search space [3]. This type of creativity challenges the divide-and-conquer paradigm dominant in software engineering – few existing methods and techniques encourage ideas to be combined in unfamiliar ways. Composition-based technologies, for example, combine software services in familiar orchestrations already specified in business process models. However, if analysts use pre-defined rules to combine requirements already found in a pre-defined search space to create novel requirements, then we can describe the

resulting requirements process as one of combinational creativity.

To conclude, Boden's three categories of creativity enable us to refine our understanding of a creative requirements process by framing it as exploratory, combinational and transformational creativity through the introduction of pre-defined rules to search a requirements space in different ways or change the space to find novel and useful requirements. But how do we define and measure novelty? The next section reports previous creativity research that addresses this challenge.

B. Defining Novelty

Defining and measuring what is novel and therefore creative is a core question for creativity research. Ritchie [51] proposed a framework that contained formal criteria that rate the extent to which a software program had behaved creatively, including the criterion *novelty*, which defines to what extent the produced item is dissimilar to existing examples of a domain. Therefore, based on this criterion, we can argue that a requirement is novel if it is dissimilar to reference requirements or other artifacts in the particular situation or domain based on the definition of situated creativity [58].

One way to do this is to leverage similarity matching algorithms developed originally in requirements research to retrieve similar requirements and models that can be reused to improve the completeness and correctness of specifications [36, 56]. In simplistic terms the measurement of creative outputs from requirements processes necessitates the opposite – algorithms that compute what is dissimilar based on formal criteria expressed as dissimilarity measures between partial and complete requirements in different exploratory, combinational and transformational search spaces. So far we have prototyped and evaluated an algorithm to measure different types of dissimilarity, such as literal and analogical, between requirements and other artifacts expressed as natural language [66]. This algorithm demonstrates that measuring requirement novelty is possible. However, more research is needed to develop and evaluate a complete set of formal criteria for the definition of a novel requirement.

C. Incubation and Illumination in Creativity

Poincare developed a philosophical model of creative problem solving based on personal reflections about his own scientific processes [23]. The model distinguished between incubation and illumination. Illumination is often a relatively short period in which ideas suddenly emerge, and is often linked to idea finding techniques. But what is just as interesting from a requirements perspective is the incubation process that is an important pre-requisite to illumination. It includes relaxing periods during which people combine ideas with a freedom that denies linear and rational thought, often away from the immediate problem solving task.

But how can requirements projects encourage and support incubation activities when resources are tight? How many clients will pay for external consultants to explore different types of search spaces not directly related to the original problem? Not many. However, De Marco [12] argues that software projects need slack – slack to enable incubation. He traced the lack of slack to management trends for down-sizing, restructuring, and cost-cutting in the name of efficiency and global competition. He argues that the resulting costs in human capital – stress, pressure, and over-commitment – often become barriers to success due to the lack of slack needed to provides analysts with the quality time to solve problems effectively and creatively.

Therefore, we argue that slack is a pre-requisite for incubation and, as a consequence, creative requirement processes. Requirements projects need to resource and plan for slack and incubation to deliver more creative requirements.

D. Different Types of Creative Process

Creativity research has also identified three basic categories for describing creative processes and their characteristics:

- Inspirationalist creative processes focus on the interplay between consciousness and unconsciousness, opportunistic insight and associated breakthrough leading to unexpected discoveries of new knowledge. The Wallas [62] model is a leading inspirationalist model, and was refined by Hadamard [54];
- Structuralist processes support rational techniques to explore and solve problems [10, 50]. Structuralist creative processes are defined by the deliberate generation and evaluation of ideas;
- Situationalist processes support the social aspects of creative problem solving. Shneiderman [54] defines four key phases: (i) collecting information from public domain and available digital sources; (ii) relating, interacting and consulting with colleagues and teams; (iii) creating, exploring, composing, and evaluating solutions; and (iv) disseminating and communicating solutions in a team and storing them in digital sources.

We argue that an effective creative requirements process should have characteristics of all three categories of creative process. It needs to have techniques that deliberately support different types of creativity in the form of pre-defined rules to search and change a space and associate elements in the space of requirements to discover novel and useful ones. In the next section we report a selection of creativity techniques that deliver such structuralist processes. The process also needs to support the collection and interaction of information to generate ideas between analysts and other stakeholders – a process that is situationalist. In section VI we report software tools that support collaborative creativity. And the process also needs to support incubation and reflection that is a pre-requisite for problem and idea finding. In section VII we report different training

requirements and techniques needed to give analysts the competencies to undertake effective creative requirements processes.

V. CREATIVITY TECHNIQUES

The original creative problem solving methods such as CPS and Synectics came with a set of creativity techniques that others have built on. Even a quick search will reveal 100s of different creativity techniques with which to find novel ideas [39], few of which are likely to have been used in many requirements projects.

Many of the creativity techniques share important characteristics that distinguish them from established requirements acquisition techniques. They seek to find large numbers of often incomplete ideas rather than precisely acquire complete information, explore search spaces rather than document requirements, and encourage stakeholder participation rather than individual analysis. Consider the following examples that support exploratory, combinational and transformational creativity.

A. Exploratory Creativity Techniques

Some established exploratory creativity techniques have already found a role in requirements processes, for example *snowballing* evolved into the *KJ-method* [49] and others such as *traditional brainstorming* and *sticking dots* are now subsumed into methods to generate and prioritize requirements. There are, however, still many other techniques that are available to enhance creative thinking in requirements processes.

Free association is an idea finding technique that relies on a stakeholder's mental stream of consciousness and network of associations [60]. In *serial association* the stakeholder starts with a given trigger and travels along a network of associations by saying or writing down the streams of ideas that come into your mind until one idea is useful. We are not aware of many uses of *free association* in requirements projects, but believe that it can be used effectively without much training in small stakeholder groups. It can also be used to overcome acquisition bottlenecks with other techniques.

Stories have an established role in requirements projects as use cases and scenarios, however *story writing* is different because its enables stakeholders to express ideas that they find otherwise difficult to express [60]. Stakeholders are encouraged to find or create a fictional parable or story with parallels to the current problem that can reveal stakeholder motives, anxieties and frailties not acquired using other techniques. *Story writing* can be seen as a natural extension of use case and scenario analyses, but its focus on creative fiction and active story telling enables the resulting stories to find new ideas in a way that use case analysis does not support.

People cannot just turn idea finding on when needed, so extend the situationalist requirements process to capture creative ideas wherever these come from. One more extreme

technique is to *keep a dream diary*, as many people experience creative insights in their dreams [38]. Although most dreams are forgotten, we can train stakeholders to come into better contact with their dreams. Capturing dreams in notebooks and recordings over time can lead to recognition of recurring themes and ideas that can be applied in projects. Although contentious, we only see *keeping a dream diary* as a natural extension of cultural probes [17] with which to capture ideas in the work and home places.

B. Combinational Creativity Techniques

Combining existing ideas involves taking different pairs of existing ideas and combining them systematically to find new ideas. One variation is *fixed and random elements*, in which one element of the problem – the fixed element – is combined with stimuli that are selected randomly taken from general information sources such as the Internet or more problem-specific sources such as domain models. Although predominantly combinational, this technique also explores and collects information with which to combine with the original stimuli.

Selecting multiple random stimuli is a technique well-suited to requirements expressed in text form. Select two or three grammatically appropriate random stimuli, such as a noun, verb and adjective, to find bizarre phrases from which to find new ideas and requirements. The degree of novelty can be controlled by selecting more stimuli from the problem domain. This technique can be applied easily to structured requirements in repositories, randomly selecting stimuli from external sources or other requirements in the repositories. It is also simple to automate, either with a restricted project lexicon or through existing on-line random phrase generators.

C. Transformational Creativity Techniques

Assumption surfacing [42] seeks to make underlying assumptions more visible. We believe that it is well-suited to requirements projects because of the importance of domain assumptions in techniques such as satisfaction arguments [24]. *Assumption surfacing* encourages an analyst to list not only assumptions that support this requirement, but also counter-assumptions that are the opposite poles of the constructs represented. The analyst then works through all assumptions and counter-assumptions, removing pairs that make little difference to support for the requirement. Remaining pairs can then be assessed for their impact on each requirement and their degree of plausibility based on evidence for the original assumption. Although requirements methods like REVEAL document domain assumptions, few if any treat finding assumptions as an active search process that can generate opportunities for transformational creativity.

Another technique is *boundary relaxation* [42]. It uses pre-defined checklists of possible boundary conditions to find opportunities for transformational creativity.

Conditions in current checklists include approval (e.g. *authorization*), resources (e.g. *money*), prior investments (e.g. *established structures*), acceptability (e.g. *tolerable levels of change*) and involvement (*actor roles*). Most existing requirements notations, for example data flow diagrams and i^* , define problem boundaries in terms of actor roles rather than other types of boundary conditions. Therefore the boundary condition checklist can provide pre-defined rules to explore conditions that can be relaxed systematically to generate opportunities for transformational creativity.

VI. CREATIVITY TECHNOLOGIES

Compared with the large number of creativity techniques identified there are fewer creativity support tools available to use, but their number has grown substantially since the 1990s. In our work we have sought to integrate creativity support and requirements tools in unfamiliar ways. In *CRIS* we integrated our ART-SCENE tool for scenario-based requirements discovery with *combinFormation*, a tool that supports people to create new requirements while finding and collecting information [30]. We also designed *AnTiQue* to retrieve analogical web services to support creative thinking about new requirements [66]. There are many similar creativity support tools to work with, so in this section we report tools to support creative thinking by individuals and teams.

Successful creativity support tools are expected to support pain-free exploration and experimentation, engagement with content to promote active learning and discovery, search retrieval and classification [21]. Many creativity support tools have these capabilities, in stark contrast to most requirements tools. Some are available commercially. For example *MindView* allows a user to explore visual mind maps and *Thought Office* supports exploratory creativity through on-line brainstorming, stream of consciousness and lateral thinking tools.

More advanced research-based tools are also available. *combinFormation* is a mixed-initiative tool that integrates searching, browsing and exploring information [31]. Software agents procedurally extract clippings from documents retrieved by search engines, and assemble them in a visual composition space storyboard. The visual composition is generated over time and related surrogates are automatically clustered. An example is shown on the left-hand side of Figure 1. The user engages in processes of searching, browsing, collecting and authoring media in the composition space, which serves as a visible medium for communication between human and agent, as well as for thinking about and sharing information resources.

Elsewhere Baumer et al. [2] reported a technique for analyzing textual corpora to identify potential metaphors with which to foster creative thinking. Evaluations reveal that the technique led to more creative outcomes using detected metaphors. We conjecture that the same technique could be applied in requirements projects to discover and

use novel metaphors with requirements problems in problem and idea finding.



Figure 1. The left-hand side shows an example visual storyboard created in combinFormation [31]. The right-hand side shows a digital table and wall [25]

Collaborative support tools are also available. Although Schneiderman [54] reported that illumination moments when ideas are found are often personal, but the processes that lead up to them are highly collaborative. A recent addition is *BootB*, a creativity support tool that provides a creativity marketplace for problem owners and creative talent to meet and collaborate. More research-based tools are also available. For example Hilligies et al. [25] developed new technologies to enhance exploratory creativity techniques such as brainstorming without losing the advantages of face-to-face communication. Their application combines an interactive table and a large digital wall display that enables active engagement with brainstormed content, shown on the right-hand side of Figure 1. The *EDC* tool reported by Warr & O'Neill [63] supports interaction with external representations to facilitate shared understanding and common ground between stakeholders engaged in creative problem solving.

VII. CREATIVITY TRAINING

Most established creativity training courses are based on the CPS, Synectics and TRIZ methods. The primary focus of most training is on creativity techniques that support deliberate and systematic idea finding based on a structuralist process. TRIZ training offers both hand-on experience with creativity techniques and new skills with which to develop patents for new ideas to enable more strategic market growth. Other training, such as with Synectics, supports a situationalist process with a stronger focus on facilitation skills to find problems and new ideas and user listening skills as one means of encouraging incubation activities, albeit in facilitated workshops. It has a different focus to most existing requirements training. We believe that analysts can benefit from training in these established creativity techniques.

VIII. AN INITIAL AGENDA FOR FUTURE RESEARCH

This vision paper is underpinned by two important assertions – that many requirements engineering processes are creative problem solving processes, and that these

processes can be enhanced to produce more useful and novel outcomes if treated as creative problem solving. We believe that we have reported arguments and evidence to support both. Our mapping process revealed many similarities between the problem understanding and idea generation phases of creative problem solving and established requirements processes. It showed that creativity techniques and software tools can be applied to fill the gap and find new ideas expressed as requirements. We used creativity categories [3] to explain why current requirements techniques and software tools do not support idea finding, describe how to extend these techniques and tools, and measure requirements artifacts for their degree of novelty. We reported existing techniques and tools that requirements projects can apply to support idea finding. In the remainder of this paper we outline a research agenda to establish idea finding in requirements engineering research and practice.

Basic research is needed to develop new models of requirements engineering that incorporate idea finding. One role of the models is to define and recognize a creative requirement in the set of information artifacts generated during a requirements process. A second is to describe the processes with which stakeholders can effectively generate creative requirements from idea finding. Naturally we encourage researchers to think creatively about models, and hope that the framework that follows can provide a sound foundation for model development.

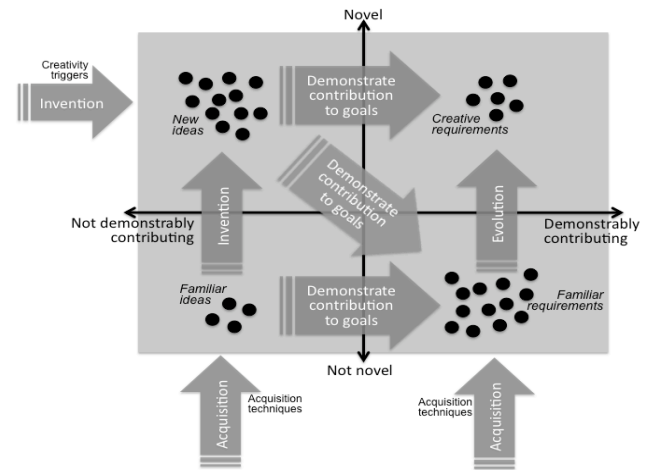


Figure 2. The space of artifacts generated in creative requirements engineering process

The space of information artifacts generated in a creative requirements process can be depicted graphically in Figure 2. Each artifact can be categorized as a requirement or idea that is more or less novel and more or less useful [57]. The vertical *novelty* dimension describes how similar an artifact is to other artifacts in the domain [58]. The horizontal *contribution* dimension equates usefulness to the contribution to goal attainment that must be demonstrated if a requirement is satisfied. Any artifact has the potential to be useful, but it is not useful until a contribution to goal attainment has been demonstrated.

Evidence suggests that most requirements generated in projects are similar to existing artifacts in the domain and not novel [29]. We depict these requirements in the bottom-right quadrant. An artifact that is both not novel and not useful describes a familiar idea for which the contribution to goal attainment has yet to be demonstrated. In contrast an artifact that is both useful and novel is characterized as a creative requirement, whilst a novel artifact not yet demonstrated to contribute to goal attainment is a new idea still to become a creative or familiar requirement.

Of course the degrees of novelty and usefulness of each information artifact are not fixed because stakeholders can change them as depicted in Figure 2. Evidence from our projects revealed that creative requirements are rarely generated directly, but result from transformations applied to artifacts acquired directly from stakeholders and other sources and often less novel and/or useful [29, 32]. Therefore, as well as the need for more idea finding, other activities are also needed to generate creative requirements from new ideas.

To turn familiar requirements into creative ones we need techniques to add novel behaviors and qualities – techniques that mutate a required behavior or quality to become novel in exploratory, transformational and combinational ways whilst still contributing to goal attainment. New research is needed to develop these techniques. In contrast, to turn novel ideas into creative requirements, we need new techniques and tools which with to match and filter ideas that do not contribute to goals expressed with techniques such as i^* [64] and KAOS [61].

New techniques need to be underpinned with extended requirements theories that describe the space of information artifacts. For example Jackson's separation of machine and domain [26] can be extended to express the dimensions of usefulness and novelty of machines and domains, and made operational as a computational model of search that equates novelty with dissimilarity [51]. We can build these models from existing applications of search-based optimization applied to requirements spaces [14].

Applied research is needed to extend requirements processes with creativity techniques that support idea finding. One challenge is to make requirements practitioners aware of the processes, techniques and tools available for idea finding. To deliver this we need a rigorous classification of creativity techniques based on a creative requirements process.

Another applied research direction is software tools. Creativity support tools already exist, and requirements researchers should integrate these tools to explore whether the resulting capabilities can foster problem and idea finding leading to the specification of more creative requirements. Requirements techniques and tools can be extended with new pre-defined rules that will encourage exploratory, combinational and transformational creativity. For example we can extend scenario walkthrough techniques in ART-SCENE with new rules to find novel as well as useful

requirements from alternative course events, and goal modeling in KAOS with new boundary condition rules that generate new types of obstacles that invalidate agent goals, to encourage transformational creativity.

REFERENCES

- [1] Apple, 2009, <http://www.apple.com/iphone/apps-for-iphone/>
- [2] Baumer E.P.S., Tomlinson B., Richland L.E. & Hansen J., 2009, 'Fostering Metaphorical Creativity using Computational Metaphor Identification', *ACM Proceedings Creativity and Cognition*, 315-324.
- [3] Boden M.A., 1990, *The Creative Mind*, Abacus, London.
- [4] Boehm B, Grunbacher P. & Briggs R., 2001, 'Developing Groupware for Requirements Negotiation: Lessons Learned', *IEEE Software* 18(3), 46-55.
- [5] Buschmann F., Meunier R., Rohnert H. & Sommerlad P. & Stal M., 1996, 'Pattern-Oriented Software Architecture: A System of Patterns' John Wiley & Sons.
- [6] Cao L. & Ramesh B., 2008, 'Agile Requirements Engineering Practices: An Empirical Study', *IEEE Software*, 25(1), 60-67.
- [7] Castro, J., Kolp, M., Mylopoulos, J., 2002, 'Towards Requirements-Driven Software Development Methodology: The Tropos Project', *Information Systems* 27(6), 365-389.
- [8] Cox 2005, Cox Review of Creativity in Business: building on the UK's strengths. at http://www.hm-treasury.gov.uk/coxreview_index.htm
- [9] Csikszentmihalyi, M., 1996, 'Creativity: Flow and the Psychology of Discovery and Invention', Harper Collins.
- [10] Daupert, D., 2002, 'The Osborn-Parnes Creative Problem Solving Manual', at www.ideastream.com/create.
- [11] de Bono, E., 2007, 'How to Have Creative Ideas', Vermilion.
- [12] DeMarco T., 2002, 'Slack: getting past burnout, busywork, and the myth of total efficiency', Broadway Books.
- [13] DTI 2005, Creativity, Design and Business Performance, DTI Economics Paper No.15, UK Department of Trade and Industry, November 2005.
- [14] Finkelstein A.C.W.F., Harman M., Mansouri A., Ren J. & Zhang Y., 2008, "'Fairness Analysis" in Requirements Assignments', *Proceedings 16th International Requirements Engineering Conference*, IEEE Computer Society, 115-124.
- [15] Floyd, C., Mehl, W.-M., Reisin, F.-M., Schmidt, G., & Wolf, G., 1989, 'Out of Scandinavia: Alternative Approaches to Software Design and System Development', *Human-Computer Interaction*, 4(4), 253-350.
- [16] Franch X. & Maiden N.A.M., 2003, 'Modelling Component Dependencies to Inform their Selection', *Proceedings 2nd International Conference on COTS-Based Software Systems*, Lecture Notes on Computer Science LNCS 2580, Springer-Verlag 81-91.
- [17] Gaver B., Dunne T., Pacenta E., 2006, 'Design: Cultural Probes', *Interactions* 6(1), 21-29.
- [18] Goldin L. & Berry D.M., 1997, 'AbstFinder, A Prototype Natural Language Text Abstraction Finder for Use in Requirements Elicitation' *Automated Software Engineering*, 4(4), 375-412.
- [19] Gordon W.J.J. 1960, *Synectics*, Harper and Row, New York.
- [20] Green T.R.G., 1989, 'Cognitive Dimensions of Notations', *Proceedings HCI'89, People & Computers V*, Ed A. Sutcliffe & L. Macaulay, Cambridge University Press, 444-460.
- [21] Greene S.L., 2002, 'Characteristics of Applications that Support Creativity', *Communications of the ACM*, 45(10), 100-104.
- [22] Greenspan S., Mylopoulos J. & Borgida A. 1994, 'On Formal Requirements Modelling Languages: RML Revisited', *Proceedings 16th International Conference on Software Engineering*, IEEE Computer Society Press, 135-147.
- [23] Hadamard J., 1954, *The Psychology of Invention in the Mathematical Field*, Dover.

- [24] Hammond J., Rawlings R. & Hall A., 2001, 'Will It Work?', Proceedings 5th IEEE International Symposium on Requirements Engineering, IEEE Computer Society, 102-109.
- [25] Hilliges O., Terrenghi L., Boring S., Kim D., Richter H., & Butz A., 2007, 'Designing for collaborative creative problem solving', Proceeding ACM Conference on Creativity and Cognition, 127-136.
- [26] Jackson M.A., 2001, 'Problem Frames: Analysing and Structuring Software Development Problems, Addison-Wesley.
- [27] Jarke M., Bubenko Y., Rolland C., Sutcliffe A.G. & Vassiliou Y., 1993, 'Theories Underlying Requirements Engineering: An Overview of NATURE at Genesis', Proceedings IEEE Symposium on Requirements Engineering, IEEE Computer Society Press, 19-31.
- [28] Jirotko M. & Goguen J., (eds) 1994, 'Requirements Engineering: Social and Technical Issues', Computer and People, Academic Press.
- [29] Jones S.V., Lynch P., Maiden N.A.M., Lindstaedt S., 2008, 'Use and Influence of Creative Ideas and Requirements for a Work-Integrated Learning System', Proceedings 16th IEEE International Conference on Requirements Engineering, IEEE Computer Society Press, 289-294.
- [30] Karlsen K., Maiden N.A.M., Kerne A., 2009, 'Inventing Requirements with Creativity Support Tools', Proceedings 15th International Working Conference, REFSQ'2009, Lecture Notes on Computer Science LNCS 5512, 162-174.
- [31] Kerne, A., Koh, E., Smith, S. M., Webb, A., Dworaczyk, B., 2008, 'combinFormation: Mixed-Initiative Composition of Image and Text Surrogates Promotes Information Discovery', ACM Transactions on Information Systems, 27(1), 1-45.
- [32] Maiden N.A.M., Ncube C. & Robertson S., 2007, 'Can Requirements Be Creative? Experiences with an Enhanced Air Space Management System', Proceedings 28th International Conference on Software Engineering, ACM Press, 632-641.
- [33] Maiden N., Robertson S. & Gizikis A., 2004, 'Provoking Creativity: Imagine What Your Requirements Could be Like', IEEE Software, 21(5), 68-75.
- [34] Maiden N.A.M., Manning S., Robertson S. & Greenwood J., 2004, 'Integrating Creativity Workshops into Structured Requirements Processes', Proceedings DIS'2004, Cambridge Mass, ACM Press, 113-122.
- [35] Maiden N.A.M. & Rugg G., 1996, 'ACRE: Selecting Methods For Requirements Acquisition, Software Engineering Journal 11(3), 183-192.
- [36] Maiden N.A.M. & Sutcliffe A.G., 1996, 'Analogical Retrieval in Reuse-Oriented Requirements Engineering', Software Engineering Journal 11(5), 281-292.
- [37] Mavin A. & Maiden N.A.M., 2003, 'Determining Socio-Technical Systems Requirements: Experiences with Generating and Walking Through Scenarios', Proceedings 11th International Conference on Requirements Engineering, IEEE Computer Society Press, 213-222.
- [38] McKim R.H., 1980, 'Experiences in Visual Thinking', PWS Publishers Wadsworth Inc, 101-103.
- [39] Michalko M., 2006, 'Thinkertoys: a Handbook of Creative-Thinking Techniques', Ten Speed Press.
- [40] Moggridge, B., 2007, 'Designing Interactions', MIT press.
- [41] Moody D., Heymans P. & Matulevicius R., 2009, 'Improving the Effectiveness of Visual Representations in Requirements Engineering: An Evaluation of the i* Visual Notation, Proceedings 17th IEEE International Requirements Engineering Conference, IEEE Computer Society Press, 171-180.
- [42] Mycoted, 2010, <http://www.mycoted.com/>
- [43] Nokia, 2010, <https://store.oivi.com/>.
- [44] Norman D., 2009, 'Technology First, Needs Last', Essay at: http://jnd.org/dn.mss/technology_first_needs_last.html
- [45] Nye, D. E., 2006, 'Technology Matters: Questions to Live With', Cambridge, MA: MIT Press.
- [46] Nguyen L., Carroll J.M. & Swatman P.A., 2000, 'Supporting and Monitoring the Creativity of IS Personnel During the Requirements Engineering Process,' Proceedings 33rd Hawaii International Conference Systems Sciences, HICSS-33, Maui, Hawaii.
- [47] Nuseibeh B & Easterbrook S.M., 2000, 'Requirements Engineering: a Roadmap'. Proceedings ICSE - Future of SE Track 2000: 35-46.
- [48] Obsorn A.F., 1953, 'Applied Imagination: Principles and Procedures of Creative Problem Solving', Charles Scribener's Sons, New York.
- [49] Ohshiro K., Watahiki K. Saeki M., 2005, 'Integrating an Idea Generation Method into a Goal-Oriented Analysis Method for Requirements Elicitation,' Proceedings 12th Asia-Pacific Software Engineering Conference, 113-121.
- [50] Plsek, P., 1997, Creativity, Innovation and Quality. ASQ Quality Press, Wisconsin.
- [51] Ritchie G., 2001, 'Assessing Creativity', Proceedings AISB-01 Symposium AI and Creativity in Arts and Science, 3-11.
- [52] Robertson J., 2002, 'Eureka! Why Analysts Should Invent Requirements', IEEE Software 19(4), 20-22.
- [53] Robertson S. & Robertson J., 1999, 'Mastering the Requirements Process', Addison-Wesley.
- [54] Scheiderman, 2007, 'Creativity Support Tools: Accelerating Discovery and Innovation', Communications of ACM 50(12), 20-32.
- [55] Shulyak L., 2010, 'Introduction to TRIZ', at http://www.aitriz.org/articles/40p_triz.pdf.
- [56] Spanoudakis G. & Constantopoulos P., 1995, 'Integrating Specifications: A Similarity Reasoning Approach', Automated Software Engineering Journal 2(4), 311-342.
- [57] Sternberg, R. J. (Ed.), 1999, 'Handbook of creativity. New York', Cambridge University Press.
- [58] Suwa, M., Gero, J. & Purcell, T., 2000. 'Unexpected discoveries and s-invention of design requirements: important vehicles for a design process', Design Studies, 21(6), 539-567.
- [59] Uchitel S., Chatley R., Kramer J. & Magee J., 2004, 'Fluent-Based Animation: Exploiting the Relation between Goals and Scenarios for Requirements Validation', Proceedings 12th International IEEE Conference on Requirements Engineering, IEEE Computer Society Press, 208-217.
- [60] VanGrundy A.B., 1988, 'Techniques of Structured Problem Solving', 2nd Ed. Van Nostrand Reinhold.
- [61] Van Lamsweerde A., 2008, 'Requirements Engineering: From System Goals to UML Models to Software Specifications', John Wiley and Sons Inc.
- [62] Wallas G., 1926, 'The Art of Thought'. New York: Harcourt, Brace and Company.
- [63] Warr A. & O'Neill E., 2007, 'Tool Support for Creativity using Externalizations', Proceedings ACM Conference on Creativity and Cognition, 127-136.
- [64] Yu E. & Mylopoulos J.M., 1994, 'Understanding "Why" in Software Process Modelling, Analysis and Design', Proceedings, 16th International Conference on Software Engineering, IEEE Computer Society Press, 159-168.
- [65] Zave P., 1995, 'Classification of Research Efforts in Requirements Engineering', Proceedings 2nd International Symposium on Requirements Engineering, IEEE Computer Society Press, 214-216.
- [66] Zachos K. & Maiden N.A.M., 2008, 'Inventing Requirements from Software: An Empirical Investigation with Web Services', Proceedings 16th IEEE International Conference on Requirements Engineering, IEEE Computer Society Press, 145-154.